UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/001,317	10/23/2001	Michael Kowalchik	EMR-00401	2338

47653 7590 12/14/2004

DAVID E. HUANG, ESQ.
CHAPIN & HUANG, L.L.C.
WESTBOROUGH OFFICE PARK
1700 WEST PARK DRIVE
WESTBOROUGH, MA 01581

EXAMINER	
CHACE, CHRISTIAN	

ART UNIT	PAPER NUMBER
2187	

DATE MAILED: 12/14/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/001,317	KOWALCHIK ET AL.
	Examiner	Art Unit
	Christian P. Chace	2187

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 October 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-23,25,30,32,34-37 and 39-50* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-23,25,30,32,34-37 and 39-50* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 October 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *8/16/04*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

This Office action has been issued in response to amendment filed 28 October 2004. Claims 1-23, 25, 30, 32, 34-37, and 39-50 are pending. Claims 24, 26-29, 31, 33, and 38 are canceled. Applicants' arguments have been carefully and respectfully considered, but they are not persuasive in light of the instant amendment and rejections necessitated thereby. Accordingly, this action has been made FINAL, as necessitated by amendment.

Examiner also wishes to note that claim 32 is designated as "currently amended" in the instant submission. However, no amendment appears to have been made. Accordingly, examiner has interpreted this to be a typographical error, and treated claim 32 and the claims that depend upon it as "previously presented."

Information Disclosure Statement

The Information disclosure statement submitted 12 August, after the previous Office action was issued, was not considered, as the submission requires a specific statement and fee pursuant to MPEP 609. Accordingly, a copy of the PTO-1449 is attached, with the references being "lined out" to indicate they have not been considered.

Power of Attorney

A change to the power of attorney was received 22 November 2004, and has been scanned and entered into the instant case.

Claim Objections

Claim 42 is objected to because of the following informalities: It depends upon canceled claim 38. Appropriate correction is required.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-23, 25, 30, 32, 34-37, and 43-50 are rejected under 35 U.S.C. 102(b) as being anticipated by Brant et al (US Patent #5,805,787) and "The RAID Book," hereinafter TRB, offered as extrinsic evidence.

With respect to independent claim 1, a cache is disclosed in the title and abstract (#16 in figure 1). A cache is, by definition, a memory subsystem in which frequently used values are duplicated for quick access. A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory. A cache is useful between a processor and a main memory when main memory accesses are slow compared to the processor speed, because the cache memory is always faster than the main memory.

A front-end interface is disclosed in figure 1, #11, as the "host interface" in column 6, line 17.

A back-end interface is disclosed in figure 1 as #24, as the "interface to the mass data storage subsystem" in column 6, line 18.

Cache storage being formed by at least two disks is disclosed in figure 1, #16, and discussed in the abstract, for example.

A cache manager is disclosed as the storage controller in figure 1, #20. As discussed in column 4, lines 9-19, the storage controller receives data access requests (via front-end interface #11) that specify respective data storage addresses, retrieves data identified by the data storage addresses, and services at least one of the requests received at the front-end interface using data stored said at least two disks. The at least one of the requests serviced are, of course, the ones in which the cache holds the information for. As the definition of cache suggests, as discussed supra, if the cache does not have the information, the cache does not service the request *per se*, but the main memory, or larger storage does (via back-end interface #24). This is also discussed in column 4, lines 9-19. As the data stored in the at least two disks that make up the cache is stored striped in a RAID configuration, as discussed in column 5, lines 34, 36, and 44, for example. This is significant as RAID 3, for example, as mentioned in column 5, line 44, adds redundant information in the form of parity to a parallel access striped array, by definition. Striping is assigning blocks of storage in regular sequence to all of an array's disks. Therefore, Brant et al uses stores the data in the at least two disks that make up the cache by striping it in a RAID 3 configuration, for example.

The cache manager storing identification data identifying addresses within said cache storage where data is stored and the corresponding address at a back-end storage area where data is stored is inherent in a direct-mapped strategy, as discussed in column 7, lines 35-40, which discusses a certain number of bits describing the

address of the main store and a certain number of bits describing the location in the cache.

The cache manager receiving a write request to store data (examiner notes that terminology such as "configured to" may raise a question as to the limiting effect of the language in the claim) and, in response to the write request, split that data into data portions (striping) and separately store the data portions on respective disks of the cache storage and the cache manager receiving a read request to read the data and, in response to the read request, concurrently (parallel) read the data portions which are separately stored on the respective disks of the cache storage (stripes) is disclosed as RAID 5, for example, in column 5, lines 36-37, and RAID 3, in column 5, line 44.

Examiner has offered TRB as extrinsic evidence that RAID 3, for example, as recited in Brant et al as well as discussed on page xvi of TRB, includes striping, which allows parallel access to the array. Split requests being part of striping on two or more disks is discussed by TRB on page 42. Page 85 discusses the fact that striped data reads and writes are handled the same way. In addition, and perhaps most importantly, page 91 discusses that striped disk arrays can be useful for storing data collected from external sources at very high transfer rates, i.e., used as a cache for a larger memory system.

With respect to claims 2 and 15, Brant et al disclose the front-end interface comprising an interface conforming to "a protocol" is disclosed in column 6, lines 39-41, for example, where the protocol is a "SCSI-type connection[s]."

With respect to claims 3 and 16, Brant et al disclose the protocol comprising at least one of SCSI, Fibre Channel, "INFINIBAND," and IDE is disclosed in column 6, line 22, which identifies IDE, as well as line 41, which identifies SCSI and associates SCSI with the front-end interface (host interface #11).

With respect to claim 4, Brant et al disclose the disks comprising disks having platters less than 3.5 inches in diameter is disclosed in column 3, lines 44-46.

With respect to claim 5, the disks comprising disks having at least one of the following platter sizes: 2.5 inches, 1.8 inches, and 1 inch in diameter is disclosed in column 3, lines 44-46, which not only discloses the 1.8 inch diameter disk, but also states that "(or smaller)" [would work in the invention]. "Or smaller" would include the 1 inch diameter as well.

With respect to claims 6 and 19, Brant et al disclose the cache implementing a RAID scheme using the disks is disclosed in column 5, lines 34, 36, and 44, in general. RAID stands for, "Redundant Array of Independent Disks." In this case, column 5, line 59 recites, "Controller 20 can include independent paths to write data to its memory in a mirrored fashion." Mirroring is redundant storage of data. The cache being an Array is disclosed in column 4, line 15, for example. Figure 1 clearly shows separate disks, and, therefore, independent disks. Therefore, RAID is explicitly disclosed embodied in the invention of Brant et al.

With respect to claim 7, Brant et al disclose the cache performing at least one of the following operations is disclosed below. Examiner reminds applicants that as the claim language stands, only one of the following limitations are required to be

anticipated by the instant prior art of record. However, it happens that all of the following limitations are anticipated by the cited prior art of record as follows:

Requesting data from a back-end storage system (see column 6, lines 50-51);

Retrieving requested data from the [at least two] disks [making up the cache] (see column 4, lines 9-19);

Sending data to the back-end system for writing (column 6,lines 50-51);

Determining the location of back-end system data within the [at least two] disks [making up the cache] (column 4, lines 32-48).

Removing data from the [at least two] disks [making up the cache] (column 4, lines 42-44).

With respect to claim 8, Brant et al disclose the addresses specifying storage locations of a back-end storage system that includes a collection of one or more disks is disclosed in the definition of a cache, discussed supra, in that, "A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory." The back-end storage system is the main memory. The back-end storage system comprising a collection of one or more disks is disclosed in figure 1, #25 and further discussed in column 6, lines 57-58, for example.

With respect to claim 9, Brant et al disclose the requests comprising I/O requests is disclosed in column 5, line 35, which refers to I/O rates. An I/O rate must be calculated from I/O requests. Therefore, I/O requests are inherent in a system with an I/O rate being measured. Also, it is important to note that I/O is "Input/Output," which is defined as the complementary tasks of gathering data for a computer or program to

work with, and of making the results of the computer's activities available to the user or to other computer processes. This is exactly what figure 1 shows – data is gathered from the disk-based disk cache and supplied to the host.

With respect to claims 10 and 21, Brant et al disclose the data storage addresses comprising data storage addresses within an address space is inherent by definition of an address, which merely denotes a location of memory in which something is, or may be, stored. The address space is the disk-based disk cache.

With respect to claims 11 and 22, Brant et al disclose the address space comprising an address space of back-end storage is disclosed by the definition of a cache, as discussed with respect to claims 1 and 8, for example.

With respect to claim 12, Brant et al disclose a storage hierarchy associated with the various contemporary storage configurations, in order of accessing speed, in column 5, lines 10-28. Column 5, lines 29-31, recites that, "A storage subsystem that has the MB cost of disk coupled with the performance of many disks operated in parallel can fill several intermediate slots in this hierarchy." By introducing the storage subsystem at another slot of the hierarchy that has a level below it of slower access storage, the subsystem effectively becomes another cache, and it's respective address space becomes another cache's address space. An example of another level of memory that would have a slower access time might be a tape library or the like.

With respect to claim 13, Brant et al disclose the cache storage having more than one disk spindle is inherent in a duplex mirrored disk subsystem, which is disclosed in column 3, line 1. A disk spindle is an axle for mounting a disk. A duplex disk

subsystem is a system of two spindles, one of which is active while the other remains on standby, ready to take over processing if the active spindle malfunctions.

With respect to independent claim 14, receiving data access requests at the cache is disclosed in column 4, lines 32-41.

The cache having storage formed by at least two disks is disclosed in figure 1, #16, and discussed in the abstract, for example.

The requests specifying respective data storage addresses is disclosed in the title and abstract. A cache is, by definition, a memory subsystem in which frequently used values are duplicated for quick access. A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory. A cache is useful between a processor and a main memory when main memory accesses are slow compared to the processor speed, because the cache memory is always faster than the main memory.

As discussed in column 4, lines 9-19, the storage controller receives data access requests (via front-end interface #11) that specify respective data storage addresses, retrieves data identified by the data storage addresses, and services at least one of the requests received at the front-end interface using data stored in the at least two disks. The at least one of the requests serviced are, of course, the ones in which the cache holds the information for. As the definition of cache suggests, as discussed supra, if the cache does not have the information, the cache does not service the request *per se*, but the main memory, or larger storage does (via back-end interface #24). This is also discussed in column 4, lines 9-19. As the data stored in the at least two disks that make

up the cache is stored striped in a RAID configuration, as discussed in column 5, lines 34, 36, and 44, for example. This is significant as RAID 3, for example, as mentioned in column 5, line 44, adds redundant information in the form of parity to a parallel access striped array, by definition. Striping is assigning blocks of storage in regular sequence to all of an array's disks. Therefore, Brant et al uses stores the data in the at least two disks that make up the cache by striping it in a RAID 3 configuration, for example.

The cache manager storing identification data identifying addresses within said cache storage where data is stored and the corresponding address at a back-end storage area where data is stored is inherent in a direct-mapped strategy, as discussed in column 7, lines 35-40, which discusses a certain number of bits describing the address of the main store and a certain number of bits describing the location in the cache.

Receiving the data access requests includes receiving a write request to store data, where servicing at least some of the requests includes splitting that data into data portions and separately storing the data portions on respective disks of the cache storage in response to the write request, and where receiving the data access requests further includes receiving a read request to read the data, and where servicing at least some of the requests further includes concurrently reading the data portions which are separately stored on the respective disks of the cache storage to retrieve the data in response to the read request is disclosed as RAID 5, for example, in column 5, lines 36-37, and RAID 3, in column 5, line 44.

Examiner has offered TRB as extrinsic evidence that RAID 3, for example, as recited in Brant et al as well as discussed on page xvi of TRB, includes striping, which allows parallel access to the array. Split requests being part of striping on two or more disks is discussed by TRB on page 42. Page 85 discusses the fact that striped data reads and writes are handled the same way. In addition, and perhaps most importantly, page 91 discusses that striped disk arrays can be useful for storing data collected from external sources at very high transfer rates, i.e., used as a cache for a larger memory system.

With respect to claim 17, Brant et al disclose the requests comprising at least one read request is disclosed in column 4, line 25, for example.

With respect to claim 18, Brant et al disclose servicing the requests comprising retrieving data from the back-end storage (main memory) and storing the data in at least one of the disks is inherent in the definition of caching. The addresses specifying storage locations of a back-end storage system that includes a collection of one or more disks is disclosed in the definition of a cache, discussed supra, in that, "A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory." The back-end storage system is the main memory. The back-end storage system comprising a collection of one or more disks is disclosed in figure 1, #25 and further discussed in column 6, lines 57-58, for example.

With respect to claim 20, Brant et al disclose servicing the requests comprising determining whether the collection of disks currently stores the requested data is disclosed in column 4, lines 39-41, for example.

With respect to independent claim 23, a data storage system is disclosed in figure 1.

A 'back-end' storage system is disclosed in figure 1, #25. Data storage spaces all have addresses, or locations for storing data – this is inherent, by definition of an address and by definition of data storage. The addresses identifying "blocks" of storage is also inherent – a block of storage may be any size, as applicants have not limited such size in the instant claim. Therefore, examiner interprets a "block" to be one memory location.

A cache for the back-end storage system is disclosed in figure 1 as #16 as well as in the title and abstract. A cache is, by definition, a memory subsystem in which frequently used values are duplicated for quick access. A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory. A cache is useful between a processor and a main memory when main memory accesses are slow compared to the processor speed, because the cache memory is always faster than the main memory.

A front-end interface is disclosed in figure 1, #11, as the "host interface" in column 6, line 17. Receiving I/O requests that specify respective addresses of back-end storage blocks is disclosed in column 5, line 35, which refers to I/O rates. An I/O rate must be calculated from I/O requests. Therefore, I/O requests are inherent in a

system with an I/O rate being measured. Also, it is important to note that I/O is "Input/Output," which is defined as the complementary tasks of gathering data for a computer or program to work with, and of making the results of the computer's activities available to the user or to other computer processes. This is exactly what figure 1 shows – data is gathered from the disk-based disk cache and supplied to the host.

A back-end interface is disclosed in figure 1 as #24, as the "interface to the mass data storage subsystem" in column 6, line 18.

Cache storage being formed by at least two disks is disclosed in figure 1, #16, and discussed in the abstract, for example. The disks having platter diameters [of] less than 3.5 inches is disclosed in column 3, lines 45-46.

A cache manager is disclosed as the storage controller in figure 1, #20. As discussed in column 4, lines 9-19, the storage controller receives data access requests (via front-end interface #11) that specify respective data storage addresses, retrieves data identified by the data storage addresses, and services at least one of the I/O requests received at the front-end interface using data stored in the cache storage. The at least one of the requests serviced are, of course, the ones in which the cache holds the information for. As the definition of cache suggests, as discussed supra, if the cache does not have the information, the cache does not service the request *per se*, but the main memory, or larger storage does (via back-end interface #24). This is also discussed in column 4, lines 9-19. As the data stored in the at least two disks that make up the cache is stored striped in a RAID configuration, as discussed in column 5, lines 34, 36, and 44, for example. This is significant as RAID 3, for example, as mentioned in

column 5, line 44, adds redundant information in the form of parity to a parallel access striped array, by definition. Striping is assigning blocks of storage in regular sequence to all of an array's disks. Therefore, Brant et al uses stores the data in the at least two disks that make up the cache by striping it in a RAID 3 configuration, for example.

The cache manager storing identification data identifying addresses within said cache storage where data is stored and the corresponding address at a back-end storage area where data is stored is inherent in a direct-mapped strategy, as discussed in column 7, lines 35-40, which discusses a certain number of bits describing the address of the main store and a certain number of bits describing the location in the cache.

The cache manager receiving a write request to store data (examiner notes that terminology such as "configured to" may raise a question as to the limiting effect of the language in the claim) and, in response to the write request, split that data into data portions (striping) and separately store the data portions on respective disks of the cache storage and the cache manager receiving a read request to read the data and, in response to the read request, concurrently (parallel) read the data portions which are separately stored on the respective disks of the cache storage (stripes) is disclosed as RAID 5, for example, in column 5, lines 36-37, and RAID 3, in column 5, line 44.

Examiner has offered TRB as extrinsic evidence that RAID 3, for example, as recited in Brant et al as well as discussed on page xvi of TRB, includes striping, which allows parallel access to the array. Split requests being part of striping on two or more disks is discussed by TRB on page 42. Page 85 discusses the fact that striped data

reads and writes are handled the same way. In addition, and perhaps most importantly, page 91 discusses that striped disk arrays can be useful for storing data collected from external sources at very high transfer rates, i.e., used as a cache for a larger memory system.

With respect to claims 25 and 30, Brant et al disclose the cache further comprising at least one interface conforming to a protocol to allow at least one additional disk to be connected to said storage is disclosed in column 6, lines 39-41, for example, where the protocol is a SCSI-type connection(s). A SCSI interface is inherently used to connect CPU's or hosts to SCSI peripheral devices (note the plural), such as many hard disks and printers, using SCSI ports, which inherently provide logical connections between the computer and all of the devices on the SCSI bus. Therefore, inherently, a SCSI bus connection through SCSI ports allows at least one additional disk to be connected through the interface.

With respect to independent claim 32, a data storage system is disclosed in figure 1.

A 'back-end' storage system is disclosed in figure 1, #25. Data storage spaces all have addresses, or locations for storing data – this is inherent, by definition of an address and by definition of data storage. The addresses identifying "blocks" of storage is also inherent – a block of storage may be any size, as applicants have not limited such size in the instant claim. Therefore, examiner interprets a "block" to be one memory location.

A plurality of caches is disclosed in column 5, lines 29-31, which discusses that the invention of Brant et al may be put in any **multiple locations** in the hierarchy of the system. Inherently, if the Brant et al invention is placed in "back-to-back" locations, if you will, then the front end interface of one of the locations will connect to the back-end interface of the next. Indeed, this is the definition of a hierarchical memory scheme. Inherently, each level stores less than the main memory, or back-end storage system, by definition. The back-end storage system is disclosed in figure 1 as #16 as well as in the title and abstract. A cache is, by definition, a memory subsystem in which frequently used values are duplicated for quick access. A cache stores the contents of frequently accessed main memory locations and the addresses where these data items are stored in the main memory. A cache is useful between a processor and a main memory when main memory accesses are slow compared to the processor speed, because the cache memory is always faster than the main memory.

A front-end interface is disclosed in figure 1, #11, as the "host interface" in column 6, line 17. Receiving I/O requests that specify respective addresses of back-end storage blocks is disclosed in column 5, line 35, which refers to I/O rates. An I/O rate must be calculated from I/O requests. Therefore, I/O requests are inherent in a system with an I/O rate being measured. Also, it is important to note that I/O is "Input/Output," which is defined as the complementary tasks of gathering data for a computer or program to work with, and of making the results of the computer's activities available to the user or to other computer processes. This is exactly what figure 1 shows – data is gathered from the disk-based disk cache and supplied to the host.

A back-end interface is disclosed in figure 1 as #24, as the "interface to the mass data storage subsystem" in column 6, line 18.

Cache storage being formed by at least two disks is disclosed in figure 1, #16, and discussed in the abstract, for example. The disks having platter diameters [of] less than 3.5 inches is disclosed in column 3, lines 45-46.

A cache manager is disclosed as the storage controller in figure 1, #20. As discussed in column 4, lines 9-19, the storage controller receives data access requests (via front-end interface #11) that specify respective data storage addresses, retrieves data identified by the data storage addresses, and services at least one of the I/O requests received at the front-end interface using data stored in the cache storage. The at least one of the requests serviced are, of course, the ones in which the cache holds the information for. As the definition of cache suggests, as discussed supra, if the cache does not have the information, the cache does not service the request *per se*, but the main memory, or larger storage does (via back-end interface #24). This is also discussed in column 4, lines 9-19. As the data stored in the at least two disks that make up the cache is stored striped in a RAID configuration, as discussed in column 5, lines 34, 36, and 44, for example. This is significant as RAID 3, for example, as mentioned in column 5, line 44, adds redundant information in the form of parity to a parallel access striped array, by definition. Striping is assigning blocks of storage in regular sequence to all of an array's disks. Therefore, Brant et al uses stores the data in the at least two disks that make up the cache by striping it in a RAID 3 configuration, for example.

The caches having respective cache storage address space is inherent – without addresses for the cache space, the system will not know here to go for requested information, or know where to put information.

With respect to claim 34, the plurality of caches being connected in series such that the front-end interface of one of said plurality of caches is coupled to the back-end interface of another of said plurality of caches is discussed supra with respect to claim 32. A plurality of caches is disclosed in column 5, lines 29-31, which discusses that the invention of Brant et al may be put in any multiple locations in the hierarchy of the system. Inherently, if the Brant et al invention is placed in "back-to-back" locations, if you will, then the front end interface of one of the locations will connect to the back-end interface of the next. Indeed, this is the definition of a hierarchical memory scheme.

Also inherent in the hierarchical scheme discussed supra, the front-end interface of the highest level cache would inherently connect to the host, while the back-end interface of the lowest-level cache would have to connect to the back-end storage system. This again, is by definition of hierarchical memory system.

With respect to claim 35, whereupon receiving one of [the] I/O requests at said front-end interface of one of the plurality of caches, the cache manager of said one of said plurality of caches sends data corresponding to said one of [the] I/O requests to said device making said one of [the] I/O requests if data is stored on the cache storage of said one of [said] plurality of caches is inherent, as it is the definition of a cache. All the instant claim recites is that if the requested data is in the cache, the cache sends the data to the host.

With respect to claim 36, whereupon receiving one of [the] I/O requests at said front-end interface of said one of said plurality of caches, the cache manager of said one of [said] plurality of caches sends said one of [said] I/O requests to one of said back-end storage [device] and another of said plurality of caches coupled to the back-end interface of said one of [said] plurality of caches if data is not stored on [in] the cache storage of said one of said plurality of caches is inherent in a hierarchical memory system, as discussed supra. The instant claim merely recites going to the next level cache if the data is not found in the first-level cache. This is how a hierarchical system such as the one discussed supra must operate.

With respect to claim 37, the cache further comprising at least one interface conforming to a protocol to allow at least one additional disk to be connected to said storage is disclosed in column 6, lines 39-41, for example, where the protocol is a SCSI-type connection(s). A SCSI interface is inherently used to connect CPUs or hosts to SCSI peripheral devices (note the plural), such as many hard disks and printers, using SCSI ports, which inherently provide logical connections between the computer and all of the devices on the SCSI bus. Therefore, inherently, a SCSI bus connection through SCSI ports allows at least one additional disk to be connected through the interface.

With respect to claims 43, 45, 47, and 50, the cache storage providing, as an initial caching capacity of the cache storage, a first cache storage size; and wherein the cache storage provides, as a subsequent caching capacity of the cache storage, a second cache storage size in response to an addition of a new disk to the cache storage, the second cache storage size being larger than the first cache storage size is

disclosed in column 6, lines 20-30 and lines 55-57. Lines 20-30 and 55-57 discuss the fact that some or all of the disks in the array may be used as a cache. If some of the disks are used as the cache, then the size would be (total # of disks) – (mass storage disks) = first cache storage size. This size would be smaller than if all of the disks were used as a cache, which would be the claimed first cache storage size. In addition, examiner has cited as extrinsic evidence of the capacity of an array changing, TRB pages 50-51, which discusses the swapping methods employed in RAID systems such as the RAID 3 system cited in Brant et al. Also, see mention of "configured to" language addressed supra.

With respect to claims 44, 46, and 48, receiving a request from another cache device connected to the cache in series as part of a cache hierarchy in response to a cache miss at the other cache device is disclosed in column 4, lines 17-24, which discusses the hierarchical known nature of a hierarchical storage system. The cache miss involving a failure of the other cache device to provide cached data corresponding to an address of the cache storage is disclosed in column 4, lines 34-48, which discuss that if the data is not in the disk cache, then the data is transferred from the subsystem, and stored in the array as well as returned to the host. Again, this hierarchical nature is discussed in column 5, as cited and discussed supra, which states that the Brant et al array can fit into SEVERAL intermediate slots in the hierarchy shown – not just ONE slot, but SEVERAL.

Obtaining the data portions which are separately stored on the respective disks (striped) from the cache storage in response to the read request, and providing the

obtained data portions to the other cache device to satisfy the request is discussed supra with respect to independent claim 1, for example. Regardless of where the striped RAID 3 cache is in the hierarchy, if the data is contained in that cache, it is delivered to the host, and, if there is a cache "above" it in the hierarchy, to that cache as well, as is discussed in the column 4 citations of Brant et al discussed supra.

With respect to claim 49, The cache manager receiving a write request to store data (examiner notes that terminology such as "configured to" may raise a question as to the limiting effect of the language in the claim) and, in response to the write request, split that data into data portions (striping) and separately store the data portions on respective disks of the cache storage and the cache manager receiving a read request to read the data and, in response to the read request, concurrently (parallel) read the data portions which are separately stored on the respective disks of the cache storage (stripes) is disclosed as RAID 5, for example, in column 5, lines 36-37, and RAID 3, in column 5, line 44.

Examiner has offered TRB as extrinsic evidence that RAID 3, for example, as recited in Brant et al as well as discussed on page xvi of TRB, includes striping, which allows parallel access to the array. Split requests being part of striping on two or more disks is discussed by TRB on page 42. Page 85 discusses the fact that striped data reads and writes are handled the same way. In addition, and perhaps most importantly, page 91 discusses that striped disk arrays can be useful for storing data collected from external sources at very high transfer rates, i.e., used as a cache for a larger memory system.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 39-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Brant et al and TRB, as applied to the claims upon which the instant claims depend, as discussed supra, in view of Katz et al (cited in prior Office action).

With respect to claims 39-42, Brant et al and TRB disclose the limitations of the claims upon which the instant claims depend, as discussed supra.

The difference between Brant et al and TRB with respect to Katz et al is the explicit recitation of the identification data corresponding to cache locations of "deferred writes." A "deferred write" is interpreted by examiner to be a write before the start of the write operation.

However, Katz et al disclose identification data corresponding to locations of deferred writes in column 12, lines 35-50, which discuss storing the identification data before the start of any write operation.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention, having the teachings of Brant et al and Katz et al before him/her, to store the identification data before the start of any write operation because it enables

such data to be used to recover from a power failure, as disclosed in column 12, lines 29-31, for example.

Response to Arguments

With respect to applicants' arguments traversing the objection to the drawings in light of the claim language that was at issue having been removed from the claims, examiner agrees, and has removed the objection to the drawings.

With respect to applicants' mention of the provisional double-patenting rejection having been removed in the copending case upon which it was applied in the instant case, examiner agrees that this is the case. Accordingly, examiner has removed the instant rejection, but reserves the right to reassert such rejection during any future prosecution of this case, should the claim language warrant same.

With respect to applicants' argument that Brant et al does not teach or suggest the instantly amended claim language, examiner believes this argument has been addressed supra with respect to the rejection of the claims to which it applies. Specifically, independent claims 1, 14, and 23 have such language added. Specifically, however, applicants cited column 7, lines 6-8 as evidence that Brant et al do not disclose a cache manager that splits data into data portions and separately store the data portions on respective disks of a cache storage. However, examiner would direct applicants the lines 59-67 of column 6, right before the instant citation, which clearly states that, "A variety of possible disk based disk cache management strategies are possible. Different management strategies will have varying impact on complexity/performance measures." Applicants are strongly urged to read the TRB

pages included as extrinsic evidence of such performance and complexity issues should any doubt as to anticipation of this claim language remain. However, even if, assuming *arguendo*, that this is not the case, the claim language remains such that each of the copies of column 7, lines 6-8 could be interpreted as a split data portion, having been "split" when copied to each disk.

It is noted herein that applicants' arguments with respect to claims 1, 14, and 23, along with the claims which depend upon them respectively, are the same, and will not be answered individually to avoid unnecessary repetition.

With respect to applicants' argument that there is no other plurality of caches between the Brant et al host and the disk system 25, examiner respectfully disagrees. Figure 1 clearly shows one cache system explicitly recited, but others are implicit, as per the discussion at column 5, line 31, which discusses that the Brant system (having the MB cost of disk coupled with the performance of many disks operated in parallel) may fill SEVERAL intermediate slots in the hierarchy shown above the instant citations. Applicants argue that this citation has been taken out of context, and is not the system of Brant et al. Examiner respectfully disagrees. The object of the Brant et al invention is to exploit the low MB cost of disk with the performance of many disks operated in parallel – indeed, this is what applicants are claiming, and what is anticipated by Brant et al, with TRB offered as extrinsic evidence of the striping inherent in RAID 3 configurations, as well as RAID 0, for example, which isn't really RAID because there is no redundancy.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Christian P. Chace whose telephone number is 571.272.4190. The examiner can normally be reached on MAXI FLEX.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald Sparks can be reached on 571.272.4201. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Christian P. Chace
Examiner
Art Unit 2187